# GES DISC

## Data and Metadata Recommendations to Data Providers


Compiled by:

Dana Ostrenga

James Johnson

Bruce Vollmer

Suhung Shen

Elliot Sherman

**Goddard Earth Sciences Data and Information Services Center (GES DISC)**

**NASA Goddard Space Flight Center**

**Greenbelt, Maryland**



Last Revised 03/31/2022 v3.4

# Revision History

| *Revision Date* | *Version* | *Changes* | *Author* |
|---|---|---|---|
| 12/09/2010 | 2.2 | Reformatted Document and cleaned up content. Added content on zonal average data. | B. Vollmer |
| 05/10/2011 | 2.3 | Updated the Metadata Table to include provenance recommended attributes for granule and collection level metadata. | D. Ostrenga |
| 6/27/2011 | 2.4 | Inserted brief section on data product testing with commonly used visualization and analysis tools. | B. Vollmer |
| 8/4/2015 | 3.0 | Renamed document title; added Table of Content, relevant resources, and a section about coordinate variables with example; Simplified Appendix A table; cleaned up content. | B. Vollmer |
| 2/21/2017 | 3.1 | Added section on Data Management to include data collections and versions. | B. Vollmer |
| 11/04/2017 | 3.2 | Added Appendix B. Made updates to links, added some clarification on shortnames and longnames. | D. Ostrenga |
| 03/24/2020 | 3.2 | Updated the Table of Contents to include subsections in the list, and to encourage NetCDF4 as the recommended format. | L. Iredell |
| 02/12/2021 | 3.3 | Updated metadata conventions | J. Su |
| 03/31/2022 | 3.4 | Update URLs to use https instead of http, and fixed outdated URLs. Document now named without any version number GES.DISC.metadata.and.data.formats. Some minor text changes. | J. Johnson |

# Table of Contents ........................................................ 3

# 1 Introduction
Improving data accessibility, increasing discoverability, and exploring applicability are the main goals for Goddard Earth Sciences Data and Information Services Center (GES DISC). To achieve these goals, interoperability is the key to the challenge and metadata is the bridge that leads to interoperability.

Data sets at the GES DISC are produced by a variety of data providers and from different institutions. To provide some consistency and uniformity across data sets, the GES DISC has established certain guidelines for metadata, data formats, and filenames for data sets archived at and distributed through the DISC.

The metadata guidelines represented in this document are an attempt to establish consistent data product requirements across subsystems, supporting the data standardization necessary for system interoperability, and are critically important for archiving and preservation to allow the datasets to be understood by both humans and the machine.

Adopting a set of conventions for metadata and data formats would (1) facilitate data management; (2) improve archiving and search processes and, thus, improve data sharing; (3) optimize the information available within the data sets; and (4) enable more data services to search, access, and use the data.

A machine independent data format that can support a rich set of accompanying metadata is the best way to maximize the information contained within the data and to maximize their accessibility and usability. The goal is to have the GES DISC data available and usable to a broad user community and be compatible with a wide array of tools (e.g., Panoply) and protocols (e.g., OPeNDAP).

Rather than providing detailed specifications, this document provides certain basic guidelines on data formats, metadata, and file naming conventions, to assist data providers in making their data products more accessible and usable to a broad user community.

# 2 Recommendations

## 2.1 Data Formats and Structure
NetCDF4 is the recommended data format. NetCDF can utilize direct access (vs. sequential access) to data, which allows efficient access to small subsets of large data sets. HDF5 and HDF-EOS5 are acceptable, but not recommended. A more detailed description of each format follows.

### 2.1.1 Formats

**NetCDF4**
The NASA EOSDIS (Earth Observing System Data and Information System) project has selected NetCDF4 as the preferred format for distribution of its standard products. NetCDF (Network Common Data Form) is a set of software libraries and data formats that supports the creation, access, and sharing of array-oriented scientific data. The format of the data is considered self-describing, i.e., there is a header which describes the layout of the rest of the file, in particular, the data arrays, as well as arbitrary file metadata in the form of name/value attributes. A NetCDF data set contains dimensions, variables, and *attributes*, all of which have both a name and an ID number by which they are identified. These components can be used together to capture the meaning of the data and relationships between data fields in an array-oriented data set. The NetCDF library allows simultaneous access to multiple NetCDF data sets, which are identified by data set ID numbers, in addition to ordinary file names. Groups, like directories in a Unix file system, are hierarchically organized to arbitrary depths. They can be used to organize large numbers of variables. Each group acts as an entire

NetCDF data set in the classic model. That is, each group can have attributes, dimensions, and variables, as well as other groups. The default root is the root group, which allows the classic NetCDF data model to fit neatly into the new model. Dimensions are scoped such that they can be seen in all descendant groups. That is, dimensions can be shared among variables in different groups, if they are defined in a parent group. Example NetCDF files and programs in a number of languages for both creating and reading NetCDF files (e.g., C, Fortran, Python, MatLab) can be found at Unidata: https://www.unidata.ucar.edu/software/netcdf/examples/files.html.

**HDF5**

HDF5 is the latest version of the Hierarchical Data Format (HDF) and overcomes many of the shortcomings of the previous version of HDF, such as not being able to store a file larger than 2 GBs and restrictive data types. The new version also addresses current and anticipated requirements of modern systems and applications. There are also many more available tools and applications for managing, manipulating, viewing, and analyzing the data in this format.

The new version follows a more comprehensive data model that includes two basic structures: a multi-dimensional array of record structures and a grouping structure. The HDF5 group is a grouping structure containing zero or more HDF5 objects, together with supporting metadata. The HDF5 data set is a multi-dimensional array of data elements, together with metadata stored in the file in two parts, a header and a data array. The storage layout can accommodate data in a variety of ways. With the default storage layout, data are stored in the same way as how they are organized in memory. Two other possible storage layouts are compact and chunked (Sec. 2.1.2).

**HDF-EOS5**

To augment the capabilities of HDF, three new data types were introduced and defined within the HDF framework: point, swath, and grid. The Point interface is designed to support data that have associated geo-location information but are not organized in any well-defined spatial or temporal way. The Swath interface is tailored to support time-ordered data such as satellite swaths or profilers. The Grid interface is designed to support data that have been stored in a rectilinear array based on a well-defined and explicitly supported projection.

These three new data types are constructed using conventions for combining standard HDF data types and are supported by a special Applications Programming Interface (API) in the application of the conventions. The APIs allow data products to be created and manipulated in ways appropriate to each data type, without regard to the actual underlying HDF objects and conventions. The sum of these new APIs comprises the HDF-EOS library.

If HD5-EOS5 is used, the geolocation field for lat/lon must be named “Latitude” and “Longitude.” For time stamps, “Time” must be used. “Time” should use the EOS convention TAI-93 given in units = “seconds since 1993-01-01.” Other variables may have any name, but it is best to use names that are as descriptive as possible and not cryptic. Common acronyms that are well understood are acceptable (such as AOD). These can be fully described using the long_name attribute of the variable.

#### 2.1.1.1 Level 3 & 4 Gridded Data File

NetCDF4, HDF5, and HDF-EOS5, with metadata following the Climate Forecast (CF) conventions (Sec. 2.2), can all represent gridded data. The HDF-EOS5 API can take care of the grid information (projection, dimensions, etc), so the data arrays only need to be stored. A standard naming convention for the data arrays, along with the units, fill value, title, and scaling information attributes, should be adopted, in order to maximize the sharing, intercomparing (with other products), and reuse of the data.

For NetCDF with CF metadata, the dimension names need to be set accordingly and projection information established manually. The NetCDF “classic” interface is simple: it allows the user to open the file, create array/variables, assign attributes, set dimensions, and close the file. (HDF-EOS5 generally has a few more steps, but the structure of the grid is set by the API.) The user must also set the longname, units, fill value, or scaling information attributes according to CF-1 conventions.

The CF conventions work best with NetCDF, because they are designed to promote the processing and sharing of files created with the NetCDF API.

#### 2.1.1.2 Level 3 Zonal Average Data File

NetCDF4, HDF5, and HDF-EOS5, with metadata following CF conventions (Sec. 2.2), can represent zonal average data. The HDF-EOS5 API supports the definition and structure of zonal average data, using the ZA object type. One of the dimensions should represent "Latitude" or zonal band.

For NetCDF4 with CF metadata, the zonal average metadata are defined by using appropriately specified dimensions. One of the dimensions should represent "Latitude" or zonal band.

#### 2.1.1.3 Level 2 Swath or Orbit Data Files

NetCDF4, HDF5, and HDF-EOS5, with CF metadata, can represent swaths.

For HDF-EOS5, the API handles the definition and structure of the swaths/orbits. Again, a standard naming convention for data arrays, units, fill/missing values, title/longname, and any scaling attributes should be adopted.

For NetCDF, the user must specify the layout of the swath structure and ensure that the relevant dimensions are appropriately designed and named. The user should adopt standard naming for data arrays and attributes (see CF convention and Standard Name Table).

Data usage and the predominant user community should guide the granularity of the Level 2 data. That is, data granules (or files) can comprise a full orbit of data (e.g., Aura OMI) or portions of an orbit or scenes (e.g., Aqua AIRS).

For full-orbit granules, one could create 15 swaths in the HDF-EOS5 swath file, to ensure an entire day's worth of Level 2 data is available.

In NetCDF4, a less orderly set-up for naming variables exists: variable1_orbitX, variable1_orbitY, etc.  Structurally, this is represented as follows:

```
/variable1_orbitX
/variable1_orbitY
/variable1_orbitZ
/variable2_orbitX
/variable2_orbitY
/variable2_orbitZ
/variable3_orbitX
/variable3_orbitY
/variable3_orbitZ
```

In HDF-EOS5, variable1 would exist in separate orbitX, orbitY, etc. swaths. Thus all the variable/arrays for each orbit can be individually accessed.

Structurally, this is represented as follows:

-/orbitX

|      /variable1
|      /variable2
|      /variable3
-/orbitY
|      /variable1
|      /variable2
|      /variable3
-/orbitZ
|     /variable1
|     /variable2
|     /variable3

### 2.1.2 Internal Gzip Compression and Data Chunking

For Level 2, Level 3, and Level 4 data, the HDF5 internal gzip compression feature is highly recommended, because it saves storage space and maximizes network bandwidth for file transfers. This feature, available through both the NetCDF4 and HDF-EOS5 and APIs, works transparently to the user. For large data granules, data chunking should also be considered. Chunking improves access to elements within the data arrays, because only the relevant part(s) of the array/chunk(s) are accessed. If the full array is accessed, all chunks are retrieved and the full array is returned to the user. Like compression, chunking is transparent to the user; the HDF5 library does all the work. Some investigation is required to determine the optimal chunk sizes.

## 2.2 CF Conventions

Earth science data sets are produced by multiple groups, utilizing various observations of the Earth system, resulting in potentially many different standards in terminology, data structure, and data format. To harmonize data sets with one another and across disciplines, the GES DISC has developed recommendations for the metadata describing these data sets that are compliant with general standards and widely-used conventions. After reviewing various metadata conventions, desired services for data, and archival practices, we have adopted the Climate Forecast (CF) conventions and standards and incorporated them into our recommendations.

One important factor in making data more interoperable and usable is the careful description of the data in standard formatted metadata. If a data set follows the conventions described below, then it is considered CFcompliant, and software and services that support CF conventions should be able to access the metadata and data from the data set.

The original purpose of the Climate and Forecast (CF) metadata conventions was to provide a clear and unambiguous standard for representing the metadata of climate and forecast model outputs encoded in the NetCDF binary format. The scope of CF metadata has subsequently broadened to include descriptions of observational data and derived products, while remaining focused on the overall design and issues of metadata representations for the earth sciences. These issues include the following:

- Data description
- Uniform identification of physical quantities
- Specification of coordinates
- Grid cell properties and interpretation

The determination of the proper conventions involved community agreement on how to specify metadata conventions for common kinds of observations, remote sensing data, and derived satellite products.

### 2.2.1 CF Metadata

The purpose of the CF conventions is to ensure that conforming data sets contain sufficient metadata to be self describing, in the sense that each variable in the file has an associated description of what it represents (long_name), including physical units if appropriate, and that each value can be located in space (relative to earth based coordinates) and time.

An important benefit of using conventions is that software tools can display data and perform operations on specified subsets of the data with minimal user intervention. Metadata describing how a field is located in time and space can be described in many different ways that a human would still recognize as equivalent. The purpose in restricting how metadata are represented is to facilitate the writing of software that parses the metadata to automatically associate each data value with its location in time and space. It is equally important that the metadata be easy for humans to write and to understand. The CF convention is designed to be backward compatible with the COARDS (Cooperative Ocean/Atmosphere Research Data Service) conventions, i.e., a conforming COARDS data set also conforms to the CF standard. Thus, new applications that implement the CF conventions will be able to process COARDS data sets.

### 2.2.2 CF Standard Names

CF Standard Names specify an agreed upon way to identify physical quantities, independent of unit or measuring method. They are not variable names but carefully constructed noun-phrases intended to be used as values of the variable specific attribute **standard_name**. CF conventions provide a searchable table of available standard names (https://cfconventions.org/standard-names.html). There are no spaces in the standard name, and the mapping between long_name and standard_name is done by discretion. If an appropriate standard name is not specified for a particular variable, then GES DISC will work with the data provider and the CF naming authority to either identify a standard name or add a new name to the CF conventions.

Some examples are:

*air_pressure_at_convective_cloud_top*
*atmosphere_net_upward_shallow_convective_mass_flux*
*mass_concentration_of_ozone_in_air*

## 2.3 Recommended Metadata for GES DISC Data Sets

The data producer should provide compliant metadata for each data set that is to be archived at GES DISC (S4PA). A data set consists of two types of compliant metadata: (1) **Collection Level** metadata that apply to all granules in a given collection and (2) **Granule Level** metadata that are specific to a given granule.

The minimum level of metadata should include the following standard attributes for all granules **(see Appendix A for tables of metadata recommendations)**:

**ShortName** - mnemonic or acronym of the data product name
**LongName** - a descriptive product name (see Section 2.4.3 for examples).
**VersionID** - collection version of the data product
**GranuleID** - name of the data granule (most commonly filename for single-file granules)
**Format** - file format of the data file
**RangeBeginningDate** - start date of the data in the file (format YYYY-MM-DD)
**RangeBeginningTime** - start time UTC of the data (format hh:mm:ss.ssssss)
**RangeEndingDate** - end date of the data in the file (format YYYY-MM-DD)
**RangeEndingTime** - end time UTC of the data (format hh:mm:ss.ssssss)
**IdentifierProductDOIAuthority** - https://dx.doi.org/ - product DOI authority
**IdentifierProductDOI** - 10.5067/xxxxxx - product DOI number

**ProductionDateTime** - the date and time that the file is generated (format YYYY-MM-DDT **hh:mm:ss.ssssssZ**, e.g., 2013-06-04T12:43:17.896000Z)
**ProcessingLevel** - The processing Level ( e.g., Level 1, 1B, 2, 2A, 2G, 3, 4)
**Conventions** – This should be an attribute with the value, e.g., "CF-1.n," where n=0, 1, 2, 3, ...

The GES DISC uses XML as the format for metadata files and uses a key set of XML attributes that are consistent with the EOSDIS Core System (ECS) data model.

## 2.3.1 Spatial Information

### 2.3.1.1 Level 3 & 4 Gridded Data (Geographic Equidistant Cylindrical Projection)

NorthernmostLatitude
WesternmostLongitude
SouthernmostLatitude
EasternmostLongitude

### 2.3.1.2 Level 2, Containing All Orbits for a Day (i.e., Global)

The Level 3 & 4 attributes may be used.

### 2.3.1.3 Level 1 or Level 2 Swath Segment of an Orbit

A set of 4 or 6 points describing the lower left, lower center (optional), lower right, upper right, upper center (optional), and upper left of the geographic polygon (aka GPolygon) describing the swath segment of the orbit is required. The points are ordered with respect to the forward flight direction of the spacecraft, running counter-clock-wise.

```
<GPolygon>
    <Point>
        <PointLongitude>  - longitude of point in the polygon </PointLongitude>
        <PointLatitude> - latitude of point in the polygon </PointLatitude>
    </Point> (repeats four or six times)
</GPolygon>
```

### 2. 3.1.4 Level 1 or Level 2 Single Full Orbit Described As Multiple Swath Segments

This requires multiple connected GPolygons. To describe, break the orbit into roughly equal-sized GPolygons by time (perhaps five-minute segments).

### 2.3.1.5 All Level 1 or Level 2 Data

In addition to the GPolygon information describing the data geographic coverage, include orbit information attributes as appropriate:

OrbitNumber
EquatorCrossingLongitude (decimal degrees)
EquatorCrossingDate (in format YYYY-MM-DD)
EquatorCrossingTime (in UTC format hh:mm:ss)

## 2.3.2 Vertical (Height or Depth) Information

Data may be 3 dimensional and hence include variables that describe the z axis. It is important to identify these variables in standard terminology, for use by applications reading and/or displaying the data. It is recommended that a description of the variable and units be provided for further clarification of the interpretation. The most commonly used variables are *level* and *height*.

In defining a variable, the following are recommended to be included:

| | |
|---|---|
| long_name | This can be something as simple as vertical level but can also be used to provide an explanation of the level. |
| Units | The valid values are listed in the [udunits.dat]() file and can be found for units of pressure, length, temperature, and density. The most commonly used are recommended. If you have a dimensionless coordinate, then it is acceptable, by convention, to use level, sigma_level, or layer; however, it is recommended then to add a standard_name attribute to describe the coordinate. |
| Positive | This is the direction of increasing coordinate values, with a valid value of either *up* or *down*. This is especially useful for data that provide the vertical axis in units that are other than a unit of pressure, such as model levels or height. If the vertical coordinate follows units of pressure, then this is not required. |

## 2.3.3 Variable Attributes

All variables (or parameters) should contain specific attributes that describe the data. These are defined in the Climate and Forecast (CF) metadata standard (https://cfconventions.org/) and can be used with NetCDF(4), HDF5, or HDF-EOS(5) variables.

### 2.3.3.1 Long_Name Attribute

The long_name attribute describes the variable as fully as possible. Sometimes, the data field variables are also given short names, such as AOD550. The long_name should expand the latter to “aerosol optical depth at 550 nm.”

### 2.3.3.2 Units Attribute (required for coordinate/dimension variables)

The units attribute describes the units of the variable stored. The name of this attribute should be “units.” For coordinate (or dimension) variables, the units attribute must also follow standards. Latitude should define units as “degrees_north” (indicating + going north), and longitude should use “degrees_east” (indicating + going east, i.e., -180 to +180 grid). (Please read more about coordinate variables in Sec. 2.3.4.). Ratios and fractions should use the units value “1”. For unitless variables, such as QA flag, the units attributes may be omitted.

### 2.3.3.3 Fill Value Attribute

The fill value attribute is automatically set and named “_FillValue” when the HDF-EOS library is used. The fill value is used where data are missing and should be outside the valid range of data.

### 2.3.3.4 Valid Range Attribute (Optional)

The “valid_range” attribute contains two elements: min and max valid value of the data. For example, cloud fraction would normally have valid range = [0.0, 1.0], but the algorithm may sometimes report values that are more or less the real valid values. “Valid_range” can be used with either NetCDF4 or HDF-EOS5 variables.

### 2.3.3.5 Scale Factor and Offset (Optional)

If the data are scaled, then the “scale_factor” and “add_offset” attributes should be included. The scaling function is:

value = scale_factor * data + add_offset, where “data” is what is stored in the file. Fill values are not scaled, so they should be set outside of the scaled valid range of data.

## 2.3.4 Coordinate Variables

CF convention supports four types of coordinates: latitude, longitude, vertical, and time. It is important to include coordinate variables, which have a “**units**” attribute that follows the standard. Because identification of a coordinate type by its units is complicated by requiring the use of an external software package [UDUNITS], it is a good practice to include standard_name. Some examples follow.

### 2.3.4.1 Longitude Coordinate

Variables representing longitude must always explicitly include the units attribute; there is no default value. The units attribute is a string formatted as per the udunits.dat file. The recommended unit of longitude is degrees_east. Also acceptable are degree_east, degree_E, degrees_E, degreeE, and degreesE.

For example:

```
float lon(lon) ;
lon:long_name = "longitude" ;
lon:units = "degrees_east" ;
lon:standard_name = "longitude" ;
```

### 2.3.4.2 Latitude Coordinate

Variables representing latitude must always explicitly include the units attribute; there is no default value. The units attribute is a string formatted as per the udunits.dat file. The recommended unit of latitude is degrees_north. Also acceptable are degree_north, degree_N, degrees_N, degreeN, and degreesN. For example,

```
float lat(lat) ;
lat:long_name = "latitude" ;
lat:units = "degrees_north" ;
lat:standard_name = "latitude" ;
```

### 2.3.4.3 Vertical (Height or Depth) Coordinate

Variables representing the height or depth dimension must always explicitly include the units attribute; there is no default value. The recommended metadata have been described in Section 2.3.2. More information can be found at https://cfconventions.org/. In the following example, an oceanographic NetCDF file encodes the depth of the surface as 0 and the depth of 1000 meters as 1000, and the axis variable would have attributes as shown.

```
float  axis(zdim) ;
axis:long_name = “Depth”;
axis:units = "meters" ;
axis:positive = "down" ;
```

### 2.3.4.4 Time Coordinate

Variables representing time must always explicitly include the `units` attribute; there is no default value. The `units` attribute takes a string value formatted as per the Unidata recommendations for the UDUNITS package. It is a good practice to include the time coordinate variable, even for a file containing just one time step. For example,

```
double time(time) ;
time:long_name = "time" ;
time:units = "days since 1990-01-01 00:00:00" ;
```

The most commonly used of these strings (and their abbreviations) includes day (d), hour (hr, h), minute (min), and second (sec, s). Plural forms are also acceptable. The date string may include date alone; date and time; or date, time, and time zone. It is recommended that the unit "years" not be used as a unit of time. Year is an ambiguous unit, because years are of varying lengths. Udunits defines a year as exactly 365 days.

# 2.4 File and Product Naming Convention

## 2.4.1 File Name

Perhaps one of the most important steps in documenting a data product is to use descriptive or self-describing file names. That is, file names should provide sufficient information for users to know what are the sensor, product, date, and version of the data they have, without them having to open the data file or refer to a separate document or metadata file. Information contained in file names should not be cryptic, and acronyms should be avoided, unless they are fairly common. Additionally, file names should be able to be parsed by a script to extract sensor, product, date, and version information.

As an example, a proper file name might look like the following:

DeepBlue-SeaWiFS_L3_v001-20091203T101431Z _20041122.he5

This file name is understandable with the proper key: “DeepBlue” stands out first as the product type, and the dash connects the product to its source, SeaWiFS in this case (other possible sources for this product are MODISAqua and MODIS-Terra). Note that “SeaWiFS” was not placed first, because that would imply this is a SeaWiFS data product. The process level, L3, indicates Level 3 grid. The version, file creation time, and date then follow. Finally, the file type (HDF-EOS5) is provided (.he5). The parts of the file name are ordered by product, level, version, and date.

The following is the file naming convention. (Note: each part of the file name, such as <product> and <level>, should not contain the underscore as delimiter.)

<product>_<level>_<version>_<begindatetime>.<suffix>, where

<product> = product type, e.g, DeepBlue-SeaWiFS

<level> = processing level, e.g., L3

<version> = version number, e.g., V001-ProductionDateTime or V001

where <ProductionDateTime> in the format of <yyyymmddTHHMMSSZ> is the file creation time.

<begindatetime> = the start datetime <yyyymm[dd[Thh[mm[ss]]Z]]>

for daily file: yyyymmdd (same format for multiple-day files, such as 8-day, 16-day)

for monthly file: yyyymm

for hourly file: yyyymmddThhZ (same format for multiple-hour files, such as 3-hourly, 6-hourly) for half/quarter-hourly file: yyyymmddThhmmZ

for orbit/swath file: yyyymmddThhmmssZ

<suffix> = data format, e.g., .nc4 for NetCDF4 or he5 for HDF-EOS5

Using this file naming convention, each part can be parsed using an **underscore** as the delimiter, and each sub-part can be parsed using the dash character. This convention also makes file names readable by humans. The "ProductionDateTime" in "version" is particularly useful to the data ingestion system, for viewing the file time creation, because files could be re-processed with the same version number (i.e., retro-processing). If the <version> in the file name does not contain <ProductionDateTime>, the data provider must notify a POC at the GES DISC regarding any retro-processing.

## 2.4.2 Variable Name

The variable (or parameter) names are not standardized by the CF conventions. We recommend data producers to follow the COARDS conventions, to ensure easy importing of data into data tools (e.g., GrADS), that is,

Variable names should begin with a letter and be composed of letters, digits, and underscores. Variable names should be case-insensitive; thus, the same case-insensitive name should not be used for multiple variables within a single file.

## 2.4.3 Data Product LongName

The data product LongName is a scientific definition of the product (or collection). It should be as brief--but complete--as possible; it expands on the more cryptic corresponding short names described in Section 2.4.4. The long name should be a character string **containing not more than 220 characters** and that is included in the file metadata as the global attribute, "LongName."

Suggest format:

*Sensor/Platform + Data subject or parameter + (Algorithm) + Processing level + Granule temporal resolution (5-min, daily, weekly, etc.) + Spatial resolution*

Deviations from the suggested format may be necessary based on the characteristics of the specific data sets. Here are some current examples:

*Aqua AIRS Level 2 Standard Physical Retrieval (AIRS+AMSU)*

*MERRA2 inst3_3d_asm_Np: 3d,3-Hourly,Instantaneous,Pressure-Level,Assimilation,Assimilated Meteorological Fields at 0.625 x 0.5 degree*

*GLDAS Noah Land Surface Model L4 Monthly 1.0 x 1.0 degree*

*SeaWiFS Deep Blue Aerosol Optical Depth and Angstrom Exponent Monthly Level 3 Data Gridded at 1.0 Degrees*

## 2.4.4 Data Product ShortName

EOSDIS originally developed the standard Earth Science Data Type (ESDT) naming convention to work with toolkits in reading data files and provide an efficient way to archive the products. The short name has continued to be beneficial in incorporating products into search and download tools and other services. The short name is an abbreviated version of the product in S4PA and has a **limit of 30 characters, where alphanumeric and "_" are the only acceptable characters**. The restriction of usage of spaces or special characters is due to various compatibility with searches and services. The short name is included in the file metadata as the global attribute, "ShortName." The short name convention should be set by the data producer, in consultation with the GES DISC POC, and documented in the product README. An example used for the MERRA data set as follows:

**ShortName convention = MCTFHVGGG**

**M = MERRA**
***C*: Configuration**, where ***C*** is one of: **A** = Assimilation; **F** = Forecast; **S** = Simulation
***T*: Time Description**, where ***T*** is one of: **I** = Instantaneous; **T** = Time Averaged; **C** = Time Independent

***F*: Frequency**, where ***F*** is one of: **1** = 1-hourly; **3** = 3-hourly; **6** = 6-hourly; **M** = Monthly Mean; **U** = Monthly Diurnal Mean; **0** = Not Applicable

***H*: Horizontal Resolution**, where ***H*** is one of: **N** = Native (2/3 x 1/2 deg); **F** = Reduced Resolution Version of Model Grid (1.25 x 1 deg); **C** = Reduced Resolution (1.25 x 1.25 deg)

***V*: Vertical Location**, where ***V*** is one of: **X** = Two-dimensional; **P** = Pressure; **V** = Model Layer Center; **E** = Model Layer Edge

***GGG*: Group**, where ***GGG*** is one of:
- **ANA** = Direct analysis products;
- **ASM** = Assimilated state variables;
- **TDT** = Tendencies of temperature;
- **UDT** = Tendencies of eastward and northward wind components;
- **QDT** = Tendencies of specific humidity;
- **ODT** = Tendencies of ozone;
- **LND** = Land surface variables;
- **FLX** = Surface turbulent fluxes and related quantities;
- **MST** = Moist processes;
- **CLD** = Clouds;
- **RAD** = Radiation;
- **TRB** = Turbulence;
- **SLV** = Single level;
- **INT** = Vertical integrals;
- **CHM** = Chemistry forcing

# 2.5 Data Management

## 2.5.1 Data Collection

Data products are organized into *collection*s of like *granule*s. A collection is a grouping of science data that all come from the same source (e.g., instrument(s), model(s), institution, etc.). Collections may also be referred to as datasets, data products or data series. Data Collections are organized by their Earth Science Data Type (ESDT), essentially the specification of an individual data product. A collection may contain zero or more granules and have common information at the dataset level across all the granules in the collection (i.e., collection level metadata).

A granule is the smallest aggregation of data that is independently managed (described, inventoried, and retrieved). Granules may be single or multiple files (single data files are most common). Granules typically represent a specific and discrete time coverage as part of the entire data record consisting in the data collection. As all data at GES DISC are available in an online archive the data granules are accompanied by corresponding metadata files that contain the essential metadata elements that support search and discovery of the data (e.g., Dataset ShortName, LongName, Version, Temporal/Spatial coverage, etc.)

### 2.5.2 Data Version

Data Version is a metadata element used to convey information on a particular variant of a data product (i.e., to distinguish between the data produced by different processing algorithms). It is used to disambiguate between different collections of the same data. To users, it is particularly useful to know which Version of a data Collection is the latest version of the Collection.

It is highly recommended to represent the data set version with an ordinal identifier (1, 2, 3 …) that expresses a position in a series. Data set version may be represented with both a major and minor version identifier (2.1). A minor version can be used to identify selected granules associated with a limited reprocessing of data (e.g., changes around a data anomaly that did not affect the rest of the collection). Whatever versioning scheme is used, it is understood that all granules in a collection were produced in a consistent manner.

The Data Version should not be mixed with other metadata attributes, either at the collection or granule level, that are intended to convey information on algorithm software used to generate the product (e.g., PGEVersion).

Periodic reprocessing of data products can produce new versions, with a distinct collection identifier. In general, data that are sufficiently different should be organized into separate collections. When data are reprocessed, the data provider must distinguish major and minor version changes and describe the nature and file/record range of every version. Something that affects the whole data set (e.g. a complete reprocessing) would be considered a major version. In practice, the data archiver may choose to combine different minor versions of data into a single major version of a collection in the archive and only advance to the next major version upon the reprocessing of the entire data collection.

Each data product version can be independent from another. That is, a L1B product may be Version 2 while a L2 product derived from L1B may undergo several algorithm iterations and reprocessings through several versions (2, 3, 4 . . . ).

### 2.5.3 Collection Identifier

A data collection is uniquely identified with a ShortName and Version. Within EOSDIS and the entities that manage data and metadata this identifier is an EntryID.

EntryID = ShortName __ Version

Some examples of current EntryIDs include:

SWDB_L2_004
AIRX2RET_006

GSSTFMC_3
MATMCPODT_5.2
ACOS_L2S_3.3
M2I1NXASM_5.12.4
*Note: the leading zeroes in the data set version are archaic and are unnecessary in the naming of future data collections.

# 3 Data Product Testing

As stated in the Introduction, the goals of these recommendations (Sec. 2) include making the data usable to a broad user community and compatible with an array of visualization and analysis tools and protocols. To meet these goals, data files provided by data producers will be tested by the GES DISC with a variety of commonly used tools, including OPeNDAP (Open-source Project for a Network Data Access Protocol), Panoply (netCDF, HDF, and GRIB data viewer), HDFView, ncdump, hdp (hdf dumper), IDV (Integrated Data Viewer), IDL (Interactive Data Language), GrADS (Grid Analysis and Display System), and Ferret (an interactive analysis tool for gridded and nongridded data that can transparently access data via OPenDAP). As data product specifications are developed for each data set, feedback from testing with these tools will be provided to the data producers and their teams, to ensure consistency with current standards and conventions and compatibility with the tools. Specific recommendations on data structures and metadata will be provided to the PIs and their teams, as appropriate.  A data provider can also test their data before soliciting feedback from the GES DISC by following instructions in **Appendix B** for some of the aforementioned tools.

# 4 Conclusion

By following a set of standards and conventions for file names, metadata, and data formats, providers of data to be archived at the GES DISC can significantly contribute to the usefulness of their data and help to facilitate data management, improve the archiving and search methods, improve data sharing, and optimize access to the information within the data.

It is recommended that the same data format for both L2 and L3 data be used, for consistency. By doing so, the data provider would not have to maintain multiple libraries. Because both NetCDF4 and HDF-EOS5 are essentially specially designed HDF5 files, users can read either format by using the HDF5 library or HDF5 tools. HDF-EOS5 is preferred, because its API takes care of the structural components of the file as grids, swaths, and point data defined by the EOS API. With NetCDF4, the CF convention must be followed, and the attributes must be set accordingly for defining swath files or projected geographic grids, in order for the files to be compatible with applications (e.g., Panoply). If a substantial portion of the user community for a given data set comprises modelers, then NetCDF is preferable, because there is significant experience in using this format within that community.

To summarize, the three key recommendations to GES DISC data providers are the following:

- Adopt GES DISC file naming conventions
- Adopt and implement CF conventions for metadata
- Make full use of NetCD4, HDF5, or HDF-EOS5 as a file format

# Relevant Resources:

CF-conventions: https://cfconventions.org/

COARDS conventions: https://ferret.pmel.noaa.gov/Ferret/documentation/coards-netcdf-conventions

NetCDF Conventions: https://www.unidata.ucar.edu/software/netcdf/conventions.html

NetCDF Example Files and Programs: https://www.unidata.ucar.edu/software/netcdf/examples/files.html

NASA Data Processing Levels:

https://science.nasa.gov/earth-science/earth-science-data/data-processing-levels-for-eosdis-data-products

CF compliance checker: https://podaac-tools.jpl.nasa.gov/mcc/

# Appendix A: Metadata Recommendations Table

The following table of metadata recommendations includes metadata that address data provenance and long-term data archival. Table rows shaded in green are required metadata, which are used by data management systems or for increasing interoperability. Newly added metadata items are in red text. Metadata items are organized in three groups:

1. Collection Level: Items are used to describe a data set and are populated as global metadata in a file.
2. Granule Level: Items are used to describe a granule or file and are populated as global metadata in a file.
3. Parameter Level: Items are used to describe a parameter/variable and are populated as SDS (Scientific Data Sets) local metadata for each parameter.

*DIF Fields: This column in the table is for use by DAAC staff to map the metadata to a DIF (Directory Interchange Format) at the Global Change Master Directory (GCMD).

## I. Global Metadata for Collection Level Information

| Metadata items | Description | Comments |
|---|---|---|
| ShortName | An abbreviated name of the product, limited to 30 characters. This should contain the version and be somewhat identifiable. A legend for determining the ShortName meaning should be included in the documentation. The shortname can only contain "_" in the name, no other non-numeric characters and be under 30 characters. The shortname should be in all capital letters. Using mixed case is discouraged. | **Required** |
| LongName | A descriptive product name, limited to 220 characters. | **Required** |
| VersionID | Version of the data set | **Required** |
| Format | Format of the data, e.g.,NetCDF-4 or HDF-EOS5. | **Required** |
| ProcessingLevel | Level of data processing, e.g., Level 1, Level 1A, Level 2, Level 3, Level 4 … | **Required (new)** |
| IdentifierProductDOIAuthority | https://dx.doi.org/ | **Required (new)** |
| IdentifierProductDOI | Product Digital Object Identifier (DOI) , format 10.5067/nnnnn, where nnnn is assigned through the ESDIS system | **Required (new)** |

| | | |
|---|---|---|
| Conventions | Value should be `CF-1 or CF-1.x (x=1,2,3,4, . . .)` | **Required** |
| source | Platforms/Instruments and any other factors related to the origin of the data product or products (if multi-sensor merged) | **Required** |
| MapProjection | Applies to gridded data. Useful for application data tools, such as ArcGIS | **(New)** |
| DataSetQuality | Overall assessment of quality of data, including relevant articles. Short summary is preferred. | **Required** |
| ValidationData | Description of or reference on how the data were validated | |
| history | An audit trail for modifications to the original data. Well-behaved generic netCDF filters automatically append their names and the parameters with which they were invoked to the global history attribute of an input netCDF file. We recommend that each line begins with a timestamp indicating the date and time of day that the program was executed. | |
| references | Published or Web-based references that describe the data or methods used to produce the data, e.g., Algorithm Theoretical Basis Document (ATBD) for the algorithm | |
| ContactPersonName | Name of the responsible person | |
| ContactPersonRole | Role of the responsible person | |
| ContactPersonEmail | electronicMailAddress of the responsible person | |
| ContactPersonAddress | Location of the responsible person or organization | |
| RelatedURL | Information about online sources related to the data | |

| | | |
|---|---|---|
| title | A succinct description of the data set<br>(similar to or same as LongName) | |
| institution | Where the data were produced | |
| ProjectAbstract | A brief summary of the project from which the data originated | |
| DataSetLanguage | Language used within the data set, LanguageCode=ISO 639 default=English | |
| ProductParameters | Keywords describing the data set | |
| Keyword | Commonly-use word(s) or phrase(s), e.g., Deep Blue Algorithm | |
| ProcessingCenter | Organization from which the data set can be obtained | |
| InputDataProducts | Input data to the product of interest | |
| InputDataProductVersion | Input data version | |
| DataProgress | Status of data set | |
| TemporalRange | Time period covered by the data set | **For GES DISC staff use** |
| SpatialCoverage | Geographic domain of the data set | **For GES DISC staff use** |
| DataResolution | Information about spatial resolution | **For GES DISC staff use** |

## II. Global Attributes for Granule Level Information

| Metadata items | Description | Comments |
|---|---|---|

| | | |
|---|---|---|
| **G1. General description** | | |
| GranuleID | For most Level 1 to Level 4 products, this is the file name. It is recommended to include descriptive information, version, and date in the file name (Sec. 2.4.1). | **Required** |
| ProductionDateTime | Date the granule was produced | **Required** |
| RangeBeginningDate | Start date of the data in the file (format YYYY-MM-DD) | **Required** |
| RangeBeginningTime | Start UTC time of the data (format hh:mm:ss.ssssss) | **Required** |
| RangeEndingDate | End date of the data in the file (format YYYY-MM-DD) | **Required** |
| RangeEndingTime | End UTC time of the data (format hh:mm:ss.ssssss) | **Required** |
| ObservationArea | Spatial coverage of the data (global or name of a geographic region) | |
| InputOriginalFile | Input data files used to generate the file | |
| ProductGenerationAlgorithm | The algorithm software used to produce the granule | |
| ProductGenerationAlgorithmVersion | The version of algorithm that generates the product | |
| OriginalFileVersion | | |
| OriginalFileProcessingCenter | The data center or project where the product is generated | |
| SpatialCompletenessDefinition | Definition of a measure of data quality: i.e. the ratio of grid elements containing valid values to total number of grid elements | **It is an evolving metadata** |
| SpatialCompletenessRatio | The data quality information: value for the previous metadata | **It is an evolving metadata** |

| **G2. Data description** | | |
|---|---|---|

| **Gridded Data ( L3 & L4)** | | |
|---|---|---|
| SouthernmostLatitude | Southernmost latitude of global grid of data set (float value) | **Required** |
| NorthernmostLatitude | Northernmost latitude of global grid of data set (float value) | **Required** |
| WesternmostLongitude | Westernmost longitude of global grid of data set (float value) | **Required** |
| EasternmostLongitude | Easternmost longitude of global grid of data set (float value) | **Required** |
| LatitudeResolution | Latitudinal grid resolution (decimal degrees) | |
| LongitudeResolution | Longitudinal grid resolution (decimal degrees) | |
| | | |
| **Swath Data (L1 & L2)** | | |
| OrbitNumber | Sequential number assigned to satellite orbits (integer) | **Required** |
| EquatorCrossingLongitude | Longitude at which the satellite crosses the equator (decimal degrees) | **Required** |
| EquatorCrossingDate | Date on which the satellite crosses the equator (format YYYY-MM-DD) | **Required** |
| EquatorCrossingTime | Time at which the satellite crosses the equator (UTC format hh:mm:ss) | **Required** |
| StartLatitude | Nadir latitude at start of swath (+ or – 90 degrees) | **Required** |
| StartDirection | Orbit direction at StartLatitude (A = ascending; D = descending) | **Required** |
| EndLatitude | Nadir latitude at end of swath (+ or – 90 degrees) | **Required** |
| EndDirection | Orbit direction at EndLatitude (A = ascending; D = descending) | **Required** |

| | | |
|---|---|---|
| NumberofOrbit | Number of orbits for the swath (needed if a file has more than 1 orbit) | |
| StartOrbit | The start orbit number (needed if a file contains multiple orbits) | |
| StopOrbit | The stop orbit number (needed if a file contains multiple orbits) | |
| FOVResolution | Field-of-view resolution of sensor used to acquire the swath data ( if multiple sensor, list FOVResolution of each senor) | |

## III. Local Metadata for Each Parameter or Variable

| Metadata items | Description | Comments |
|---|---|---|
| long_name | Parameter long name; recommend using underscore as delimiter; not recommend using dash and white space | **Required** |
| standard_name | Parameter name recommended by CF – convention community (https://cfconventions.org/) | **(new)** |
| _FillValue | Fill value or missing value (format: number, e.g., float or double, integer). Not for coordinate variables<br><br>For variables that contain two types of missing values, we recommend adding a flag variable to indicate the missing data type. | **Required (new)** |

| | | |
|---|---|---|
| units | Must follow standards for coordinate (or dimension) variables, such as longitude, latitude, and time (Sec. 2. 3.4) Examples for unitless variables:<br><br>fractions or ratio => “1” or “percent” number of pixels in a grid cell => “count” index (or flag) variable => may exclude units | Required |
| valid_range | Valid minimum and maximum values | **(new)** |
| scale_factor | Scaling factor (use for scaled variables) | |
| scale_offset | Scaling offset (use for scaled variables) | **(new)** |
| flag_values | Values of data quality flag or any other flag index variable, e.g., flag_values = 1, 2, 3,4 | **(new)** |
| flag_meanings | Meanings of flag values, e.g., flag_meanings = no_confidence, marginal, good, very_good for flag_values = 1, 2, 3, 4, respectively | **(new)** |
| comments | Additional description or comments about a parameter | **(new)** |

# Appendix B: Testing Data for Interoperability in Various Tools

Interoperability of data is a core tenant at NASA's data centers and adherence to metadata conventions will support this ask. Viewing metadata is necessary to check for metadata convention compliance. This document serves as a guide to study the metadata of NASA datasets through multiple programs. It also provides access to additional resources so the user can better understand and adhere to metadata conventions. This document specifically refers to adherence of Climate Forecast (CF) conventions. This document is intended for internal use within GES DISC.

## 2. Metadata resources and documentation

GES DISC has produced a document outlining metadata recommendations for data providers. This document summarizes appropriate data formats, structures, and CF metadata conventions. It is useful for both the data provider and the GES DISC person of contact to review this document.

- https://discette.gsfc.nasa.gov/mwiki/images/b/bf/MEaSUREs_data_at_the_GES_DISC_Recommendation_to_Data_Providers_V3.pdf

ESDIS Standards Office has extensive documentation on NASA data standards. The answers to most questions regarding data standards can be found here.

- https://earthdata.nasa.gov/standards

The Physical Oceanography DAAC has developed a web tool to check for CF, ACDD (Attribute Convention for Data Discovery), and GDS2 (Group for High Resolution Sea Surface Temperature Data Specification, version 2) compliance. The user will input the NetCDF or HDF file, and the webtool will check for the selected metadata compliance. The webtool will highlight where metadata is not compliant

- https://podaac-uat.jpl.nasa.gov/mcc/

Another CF checker is also available though the Natural Environment Research Council

- https://pumatest.nerc.ac.uk/cgi-bin/cf-checker.pl

The HDF Group has written tutorials specifically for visualization of NASA HDF files using multiple programs

- https://portal.hdfgroup.org/display/support

Metadata compliance helps with the interoperability of NASA data sets. The Dataset Interoperability Working Group (DIWG) has released a document outlining recommendations for data set interoperability of Earth Science data.

- https://earthdata.nasa.gov/esdis/esco/standards-and-references/dataset-interoperability-recommendations-for-earth-science

## 3. Metadata compliance and interoperability

This section briefly covers key aspects of metadata compliance and interoperability for data hosted at the GES DISC. Primary documents for metadata compliance and interoperability are mentioned in section 2 of this document. Please consult the GES DISC Data and Metadata Recommendations to Data Providers for a thorough explanation of metadata guidelines at the GES DISC. This section serves as a simple reference guide highlighting key aspects of metadata compliance and interoperability. It is not intended as a final check list for metadata compliance or interoperability.

**Data format:** NetCDF4, HDF5 and HDF-EOS5 are all acceptable formats, with a strong preference for NetCDF4. These data formats are intended to be self describing. To aid in the interoperability of these data sets, the self describing nature of these data formats should be utilized. The header information and attributes for data sets should be complete enough to fully describe the data. The user should not need to consult external documentation to use the data. See section 2 of the GES DISC Data and Metadata Recommendations to Data Providers document for specific details on metadata requirements.

To maximize NetCDF4 and HDF5 interoperability, HDF5 files should be accessible from the NetCDF4 API as much as possible. Section 2.1 of the DIWG document clearly outlines how this can be achieved. A simple way to validate NetCDF4 and HDF compatibility is to use the 'ncdump' command (outlined in section 4.1 of this document) and the JPL Web-based Metadata Compliance Checker (MCC). If both tools work with the HDF5 file in question, it is likely it will be interoperable with NetCDF4.

**Data format summary:**

- Check that data is in NetCDF4 (or HDF5) format
- Check that data is self describing (contains required metadata)
- Use the 'ncdump' command or the JPL Web-based Metadata Compliance Checker (MCC) on HDF5 files to check for interoperability with NetCDF4.

**Recommended metadata:** A list of the minimum level of metadata is described in section 2.3 of the GES DISC Metadata Recommendations document. These metadata need to be met before data is accepted at GES DISC. The 2016 DIWG Recommendations document presents a list of Basic CF Attributes that should be included in section 2.2. Adherence to the Basic CF Attributes recommendations, and inclusion of the minimum level of metadata will add clarity of the data.

**Recommended metadata summary:**

- Check for minimum level of metadata as described in the GES DISC Metadata Recommendations document
- Check for adherence to the Basic CF Attributes as described in the 2016 IDWG Recommendations document.

**CF Compliance:** Compliance of metadata is crucial for the interoperability and general usage of NASA data. CF compliance can be checked by using the CF checkers in section 2, or by viewing the metadata on your own. Section 4 describes how to view metadata for NetCDF and HDF files using numerous programs.

# 4. How to view metadata

This section outlines how to view metadata using multiple programs. If there is a program that should be added, or if you need any assistance viewing metadata please contact the GES DISC help desk (gsfc-dl-help-disc@mail.nasa.gov).

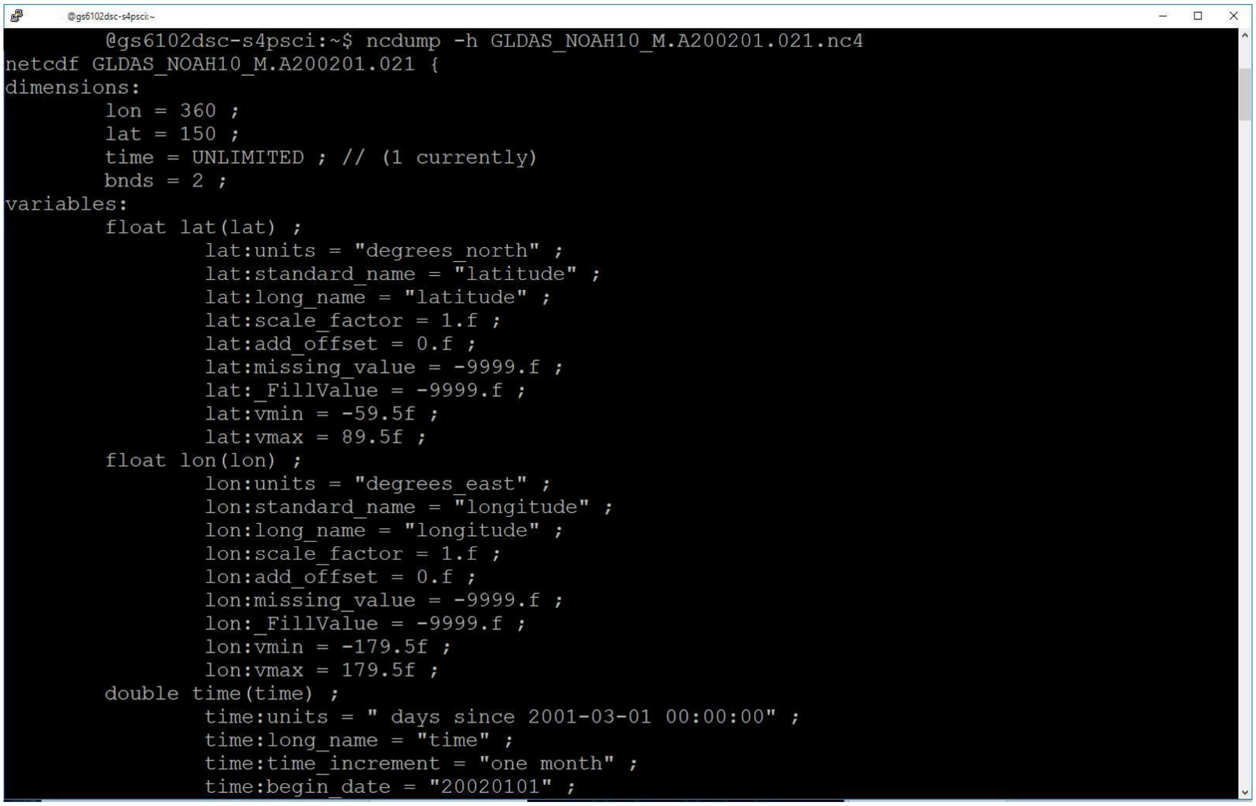
## 4.1 View metadata with NetCDF Operator 'ncdump' command

The 'ncdump' command is an easy way to view metadata and data for NetCDF, and most HDF, files in a terminal window environment. If you need to view metadata in a terminal window, this is the easiest method. ncdump is versatile and has multiple options. Visit the following link for ncdump documentation and usage: ncdump documentation. Below are simplified commands that will likely provide what metadata is needed. Using ncdump on an HDF file is also a good way to test for interoperability of the HDF dataset with NetCDF.

To view the entire contents of a file, use the following command in the terminal window:

```
ncdump file1
```

The previous command may print out much more information than is desired. The following command will print out the header information of the file, and will likely be more useful. See figure below for example. Example data can be downloaded here.  ncdump -h file1




The following command will print out the specified variable data with three significant digits.
h5dump -H filename

- ncdump -v variable1 -p 3 file1

## 4.2 View metadata with the ‘h5dump’ command

The ‘h5dump’ command is very similar to the ncdump command. They are both executed through the terminal window with the same input structure, as in “h5dump -’character’ ‘filename’”. h5dump will only work with HDF5 and HDF-EOS5 files. The h5dump command has numerous options for printing the desired data or metadata from an HDF5 file. To view these options type the following in the terminal window: ‘h5dump -h’. Below are a few simple commands to print metadata.

To view header information use the following command

- h5dump -H filename

To view attribute information use the following command

- h5dump -A filename

To list file contents use the following command

- h5dump -n filename

## 4.3 View metadata with Panoply

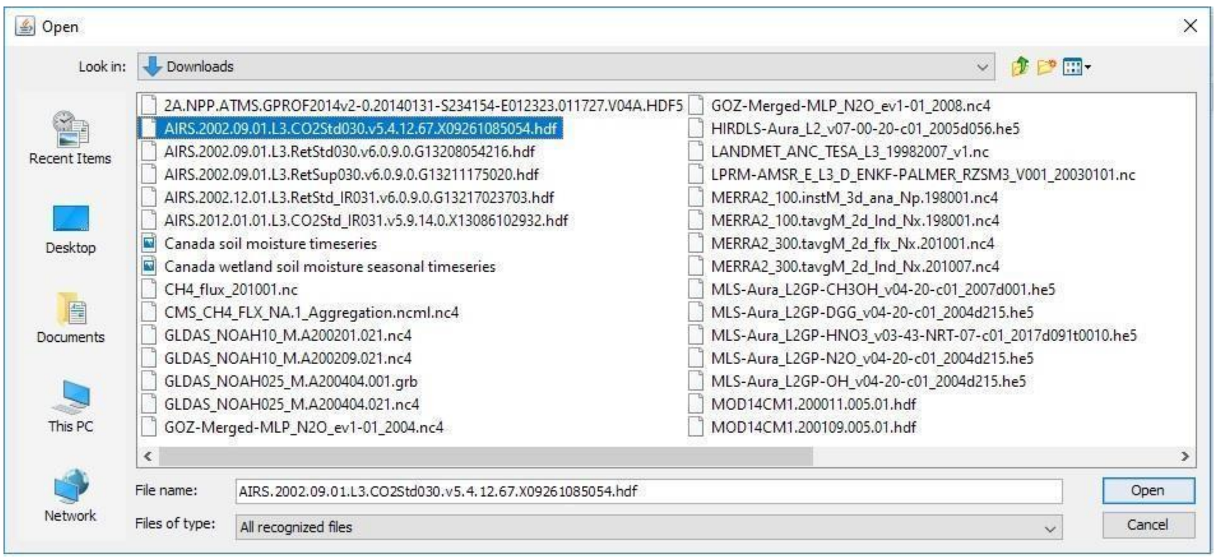
Panoply is a data viewing program developed at NASA. It accepts NetCDF, HDF, and other data formats. Panoply has numerous data visualization capabilities, but can also be used to view metadata. Panoply can be downloaded here. https://www.giss.nasa.gov/tools/panoply/

Guides to using Panoply and visualizing data can be found here

- https://www.geo.uni-bremen.de/Interdynamik/images/stories/pdf/visualizing_netcdf_panoply.pdf
- http://www.meteor.iastate.edu/classes/mt452/EdGCM/Documentation/EdGCM_Panoply.pdf

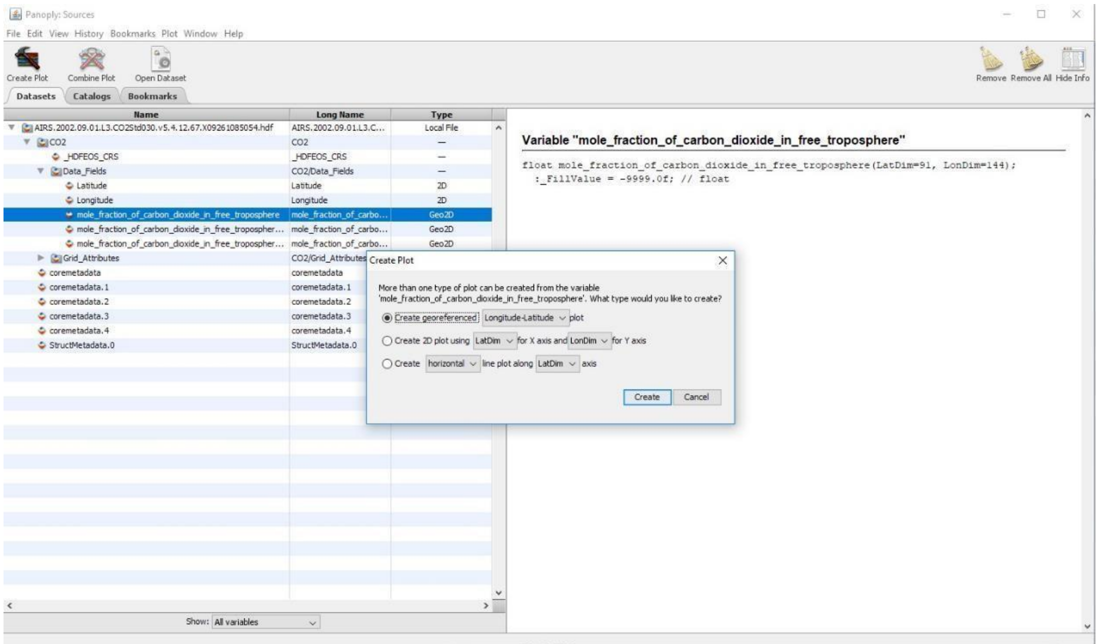
To view metadata with Panoply first install Panoply, then follow the steps below.

1. Load data into Panoply. Select data and click the 'Open' button.
   a. In this guide we are using AIRS3C2M monthly CO2 data available here: AIRS DATA

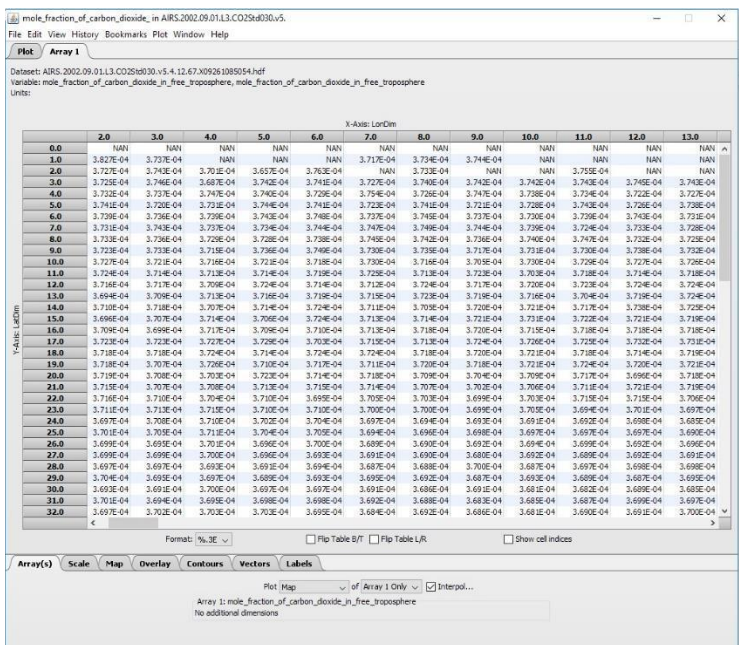



2. Once the data is loaded into Panoply, metadata will be shown on the right in the 'Info' box. Data has been loaded into Panoply if you see information under 'Name', 'Long Name', and sometimes 'Type' on the left-hand side. If data has been loaded, but is not displayed on the right, try clicking the 'Hide Info' button on the upper right.
   a. The information on the right is printed similar to the 'ncdump' command. It lists the headings within the data file.

3. To view the data, select the desired variable, then click the 'Create Plot' icon on the upper left. Select 'Create georeferenced longitude-latitude plot' then click the 'create' button.




4. This will create a plot of the data. To view the data values, click the 'Array 1' tab in the upper left. Make sure the 'Format:' menu is selected to an appropriate setting. The default is usually not sufficient to view data values.




## 4.4 View NetCDF metadata with Interactive Data Language (IDL)

IDL is a programming language mostly used for data analysis and visualization of data. It is a popular program used within the scientific community, but is also used by industries that require image processing. Syntax is somewhat similar to that of Fortran or C. This section will cover viewing NetCDF metadata with IDL.

There is no simple way to view HDF metadata with IDL. The user must write a script reading in the HDF file and perform additional operations. I would recommend the user employ 'ncdump' or Panoply to view HDF metadata, and not IDL. See links for using IDL to process HDF (https://harrisgeospatial.com/docs/HDF_Overview.html) and HDF5 (https://harrisgeospatial.com/docs/HDF5_Overview.html) data.

### 4.4.1 View NetCDF and NetCDF4 metadata with IDL

Both NetCDF (file.nc) and NetCDF4 (file.nc4) metadata can be viewed using the 'NCDF_LIST' command. This command will list variables, attributes and header information. The NCDF_LIST command has one argument, which is the input file, and multiple keywords. The syntax is as follows.

- NCDF_LIST, *filename*, /DIMENSIONS, /GATT, OUT=variable, /QUIET, /VARIABLES, /VATT, VNAME=variable

  /dimensions will print the dimensions within the file
- /gatt will print global attributes
- /vatt will print variable attributes
- /variables will print variables

If you are already familiar with IDL or working in a terminal window, the following tutorial may not be necessary. See following link for NCDF_LIST documentation: NCDF_LIST documentation

1. First the user must open a terminal window and move to their work directory. Then type 'idl' to start IDL.
2. Then type: "ncdf_list, 'file.nc', /variables, /dimensions, /gatt, /vatt". This will list variable names, dimensions, and global and variable attributes respectively.

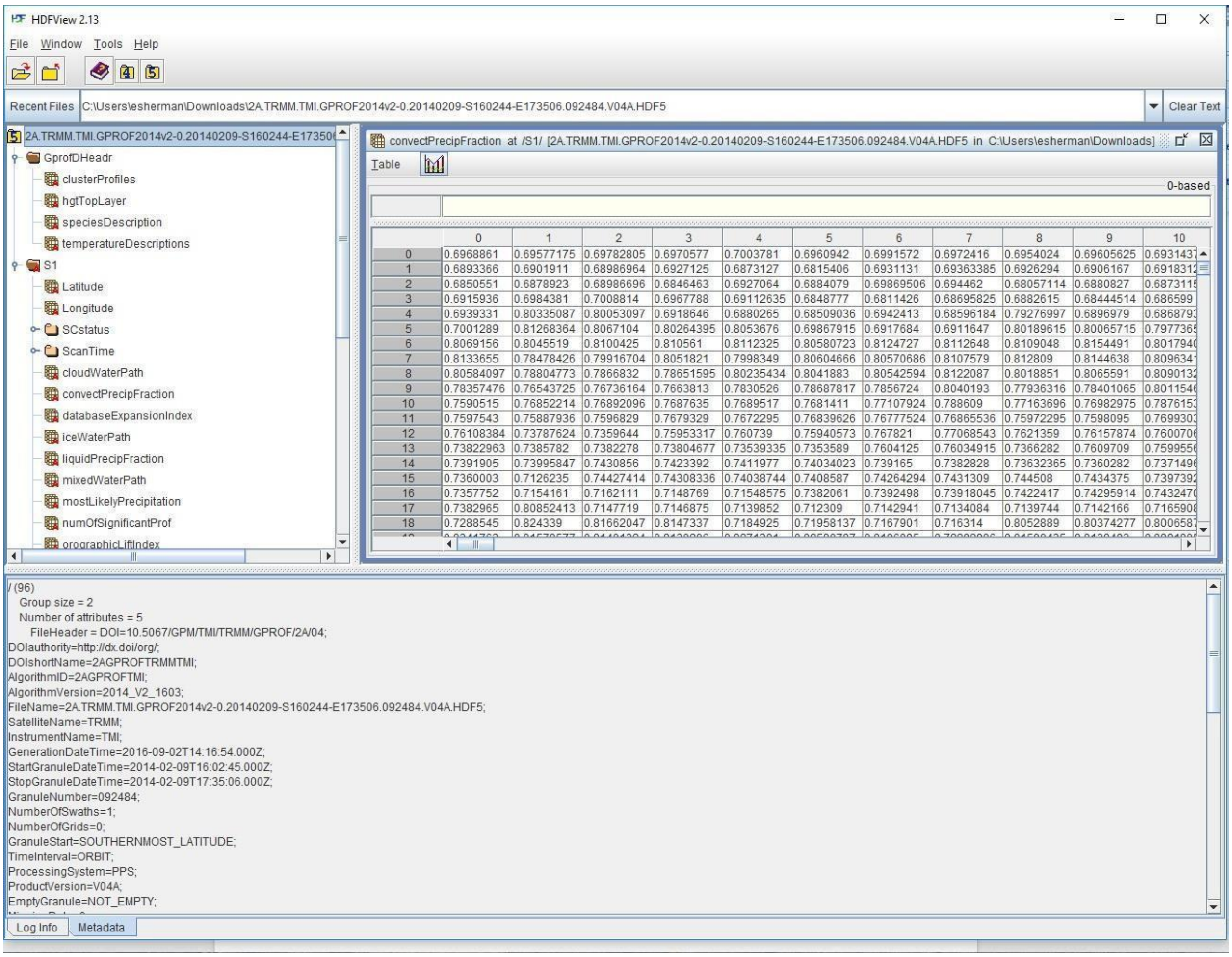
Example below:

```
IDL> ncdf_list,'GLDAS_NOAH10_M.A200201.021.nc4',/dimensions,/gatt,/vatt,/variables
% Loaded DLM: NCDF.

 GLDAS_NOAH10_M.A200201.021.nc4
     # dimensions: 4
     # Variables: 40
     # Global attributes: 16
     The unlimited dimension is 2

     Dimensions
           0  Name: lon  Size: 360
           1  Name: lat  Size: 150
           2  Name: time  Size: 1
           3  Name: bnds  Size: 2

     Global Attributes
           0  missing_value: -9999.00
           1  tavg definision:: past 3-hour average
           2  acc definision:: past 3-hour accumulation
           3  inst definision:: instantaneous
           4  title: GLDAS2.1 LIS land surface model output monthly mean
           5  institution: NASA GSFC
           6  source: Noah_v3.3
           7  history: created on date: 2016-06-29T22:26:54.129
           8  references: Rodell_etal_BAMS_2004, Kumar_etal_EMS_2006, Peters-Lidard_etal_ISSE_2007
           9  conventions: CF-1.6
          10  comment: website: http://ldas.gsfc.nasa.gov/gldas, http://lis.gsfc.nasa.gov/
          11  MAP_PROJECTION: EQUIDISTANT CYLINDRICAL
          12  SOUTH_WEST_CORNER_LAT: -59.5000
          13  SOUTH_WEST_CORNER_LON: -179.500
          14  DX: 1.00000
          15  DY: 1.00000
```

## 4.5 View Metadata with HDF View

HDFView is a tool to view and plot HDF data. Although it can sometimes read-in NetCDF data, it is only intended for HDF files. HDFView can be used to easily view metadata and data values for HDF files.

1. Open HDFView, and then open an HDF file. In this tutorial we are using the following [GPM data](GPM data).



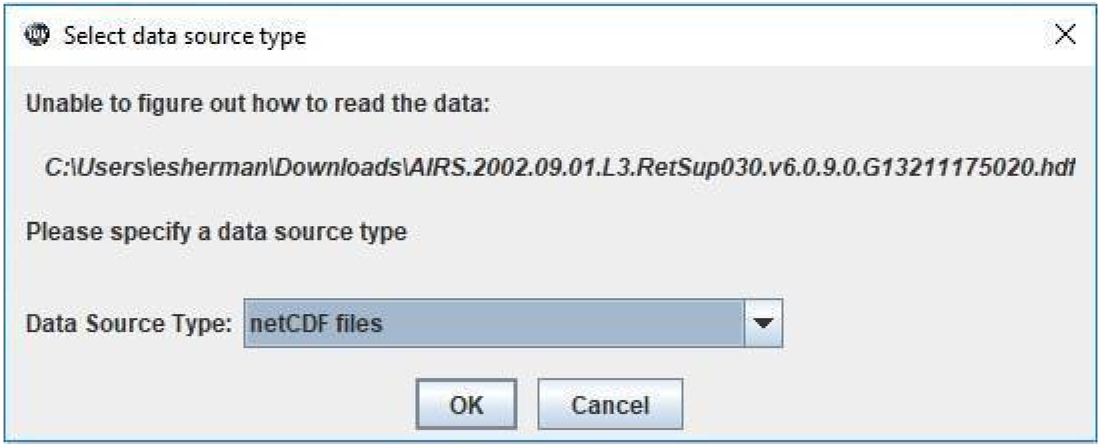

2. Metadata is displayed on the bottom panel with the 'Metadata' tab.
   a. Metadata information for each group or variable can be accessed by clicking the desired group folder or variable.
   b. Double clicking the variable will generate a matrix of data displayed on the right.

## 4.6 View metadata with Integrated Data Viewer (IDV)

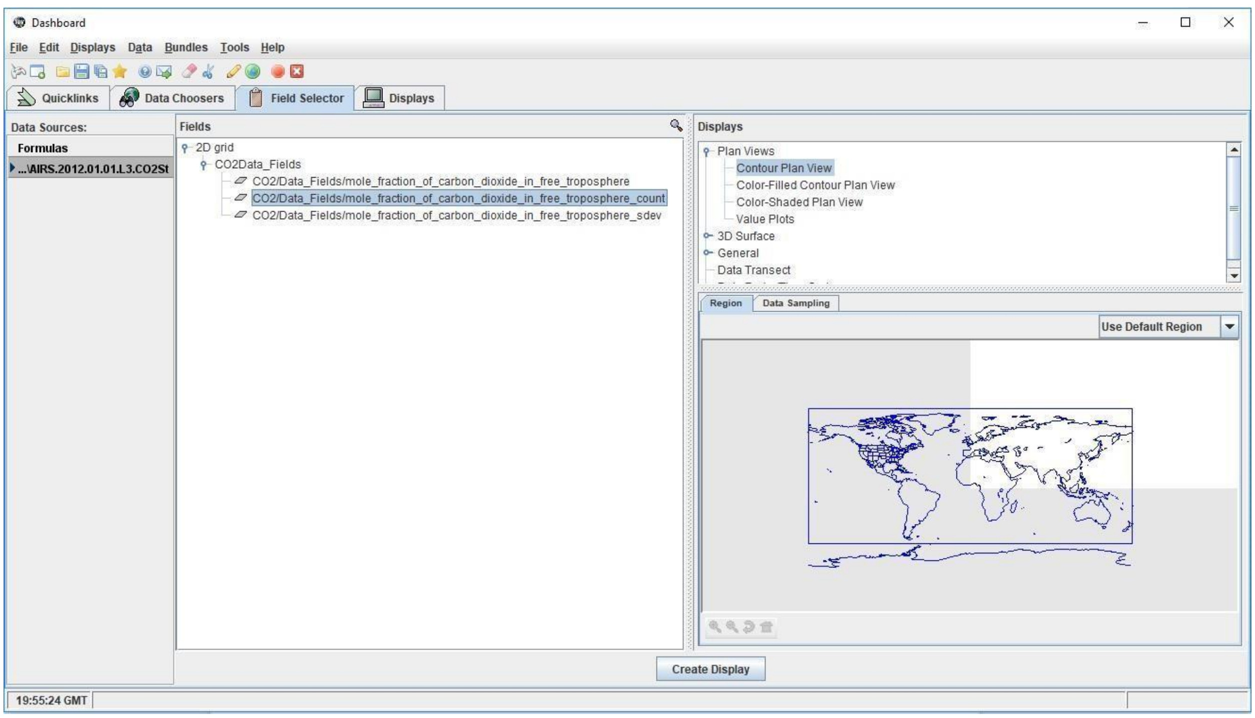
IDV is a program used for visualizing many categories of geoscience data including gridded and non-gridded data types. Although HDF file format is not included in the file read-in procedure, IDV can usually accept HDF and treat it as a NetCDF file. The procedure for reading in NetCDF and HDF files is the same and is described below.

1. Open data in IDV. This can be done by going to File -> Open or clicking the 'open file' icon on the upper left. The data for this example is [AIRS monthly CO2 data](#).

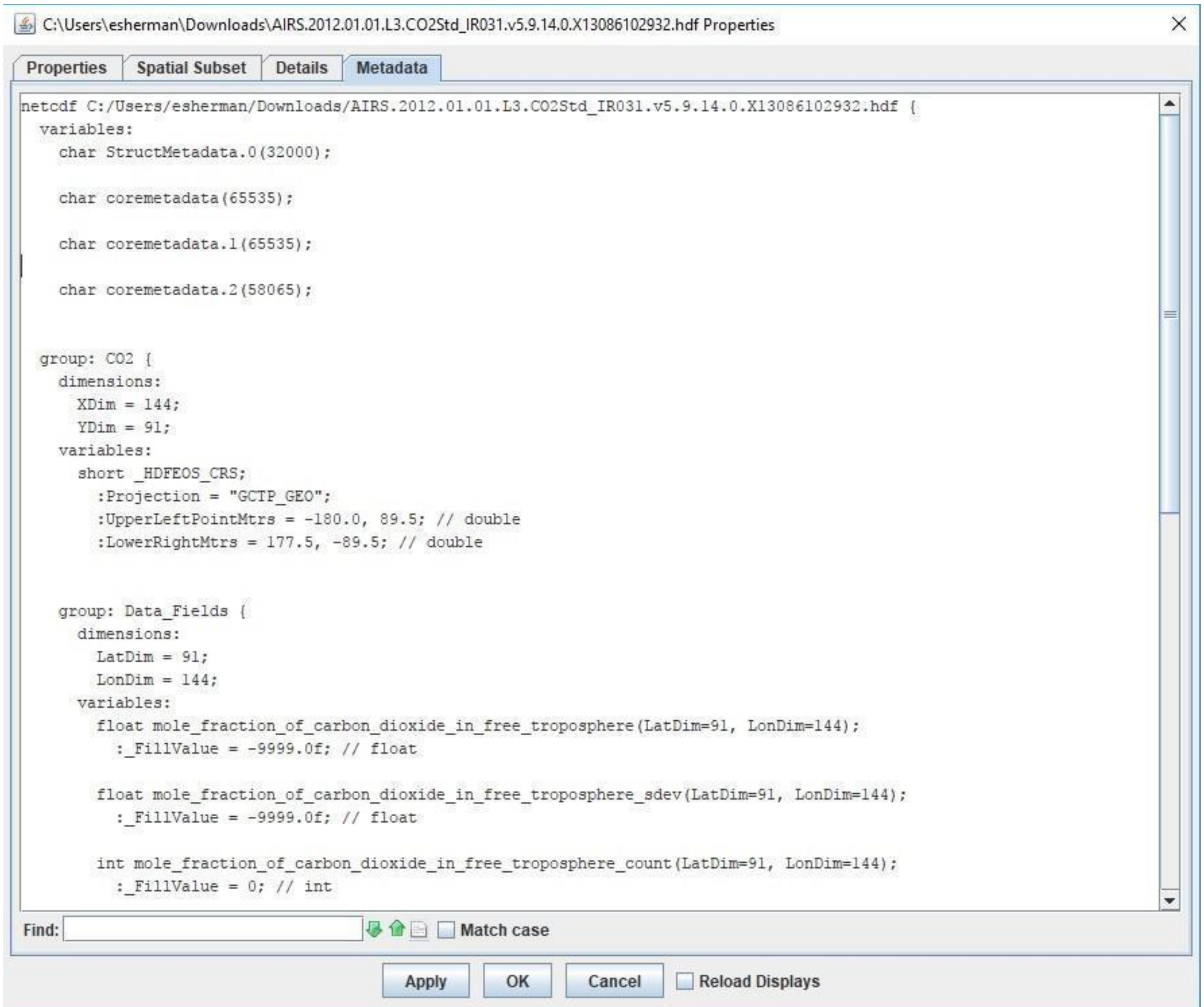
a. Choose your data file. NetCDF will not prompt 'Select data source type' window, while the HDF file formats will. If you are opening an HDF file, select the 'netCDF files' option in the 'Data Source Type:' drop down menu.




2. The AIRS CO2 data will be displayed on the left side of the the 'Dashboard' window, under the 'Field Sector' tab.




3. Right click the dataset name in the 'Formulas' box and then click 'Properties'. The metadata will be displayed under the 'Metadata' tab on the right.



C:\Users\esherman\Downloads\AIRS.2012.01.01.L3.CO2Std_IR031.v5.9.14.0.X13086102932.hdf Properties
Properties
Spatial Subset
Details
Metadata
netcdf C:/Users/esherman/Downloads/AIRS.2012.01.01.L3.CO2Std_IR031.v5.9.14.0.X13086102932.hdf {
variables:
char StructMetadata.0(32000);
char coremetadata(65535);
char coremetadata.1(65535);
char coremetadata.2(58065);
group: CO2 {
dimensions:
XDim = 144;
YDim = 91;
variables:
short _HDFEOS_CRS;
:Projection = "GCTP_GEO";
:UpperLeftPointMtrs = -180.0, 89.5; // double
:LowerRightMtrs = 177.5, -89.5; // double
group: Data_Fields {
dimensions:
LatDim = 91;
LonDim = 144;
variables:
float mole_fraction_of_carbon_dioxide_in_free_troposphere(LatDim=91, LonDim=144);
:_FillValue = -9999.0f; // float
float mole_fraction_of_carbon_dioxide_in_free_troposphere_sdev(LatDim=91, LonDim=144);
:_FillValue = -9999.0f; // float
int mole_fraction_of_carbon_dioxide_in_free_troposphere_count(LatDim=91, LonDim=144);
:_FillValue = 0; // int
Find:
Match case
Apply
OK
Cancel
Reload Displays